Integrated chance constraints: reduced forms and an algorithm

WILLEM K. KLEIN HANEVELD
MAARTEN H. VAN DER VLERK*

Department of Econometrics & OR, University of Groningen
PO Box 800, 9700 AV Groningen, The Netherlands
m.h.van.der.vlerk@eco.rug.nl

May 27, 2002

Abstract

We consider integrated chance constraints (ICC), which provide quantitative alternatives for traditional chance constraints. We derive explicit polyhedral descriptions for the convex feasible sets induced by ICCs, for the case that the underlying distribution is discrete. Based on these reduced forms, we propose an efficient algorithm for this problem class.

The relation to conditional value-at-risk models and (simple) recourse models is discussed, leading to a special purpose algorithm for simple recourse models with discretely distributed technology matrix.

For both algorithms, numerical results are presented.

Key words: chance constraints – simple recourse – algorithm

Mathematics Subject Classification (1991): 90C15

1 Introduction

Consider the linear programming problem with random constraints

$$\begin{array}{ll} \min\limits_{x} & cx \\ \text{s.t.} & T(\omega)x = h(\omega) \\ & x \in X, \end{array} \tag{1}$$

where $X := \{x \in \mathbb{R}^n_+ : Ax = b\}$ is shorthand for the usual deterministic constraints and the nonnegativity restrictions. The m random constraints $T_i(\omega)x = h_i(\omega)$, $i \in I := \{1, \ldots, m\}$, represent m different goals: we would like to satisfy each of these linear constraints. However, at the time that x needs to be decided, the actual values of the matrix $T(\omega)$ and right-hand side $h(\omega)$ are not known, because they depend on the underlying random vector ω. (We will assume that this dependence is linear). Only probabilistic information on ω is known; that is, we assume that the distribution of ω is given.

Such problems are very common in practice. For example, in the case that model (1) represents a production planning problem, $T(\omega)$ may describe unreliable technology which is used to meet uncertain demand for products $h(\omega)$. In a financial application, $T(\omega)x$ could

*The research of this author has been made possible by a fellowship of the Royal Netherlands Academy of Arts and Sciences.

stand for the random yield of a portfolio x which should (at least) meet a certain benchmark $h(\omega)$.

Model (1) is obviously not well defined. To arrive at a meaningful model, it needs to be extended. Two classical approaches used in *stochastic linear programming* to deal with deviations $\eta_i(x,\omega) := T_i(\omega)x - h_i(\omega)$, $i \in I$, are

(a) *Penalty costs*: For each individual deviation, unit penalty costs q_i^+ and q_i^- are assigned to, respectively, surpluses $\eta_i(x,\omega)^+ := \max\{0, \eta_i(x,\omega)\}$ and shortages $\eta_i(x,\omega)^- := \max\{0, -\eta_i(x,\omega)\}$; the objective function becomes

$$cx + \mathbb{E}_\omega \left[\sum_{i \in I} \left(q_i^+ \eta_i(x,\omega)^+ + q_i^- \eta_i(x,\omega)^- \right) \right].$$

The resulting expected penalty cost model is also known as the *simple recourse* model. As suggested by this name, this approach can be extended to construct more general recourse models (see e.g. the text book [2] and [22] for the closely related multiple simple recourse model). The underlying idea of this class of models is to allow infeasibilities, but to assign so called (expected) recourse costs to them.

(b) *Chance constraints*: Alternatively, one may specify a model in which feasibility is restricted. Such a model only makes sense for inequality constraints. For example, depending on the application at hand, we accept surpluses but restrict shortages by the *individual chance constraints*

$$\Pr\{\eta_i(x,\omega)^- > 0\} \leq \alpha_i, \quad i \in I,$$

and/or the *joint chance constraint*

$$\Pr\{\eta_i(x,\omega)^- > 0,\ i \in I\} \leq \alpha,$$

where α_i and α are given risk parameters in $(0, 1)$. The interpretation of such chance constraints is that a solution $x \in X$ is feasible only if it is not too risky.

Next to (mathematical) similarities, there are some important modeling differences between these two classical approaches. In particular, the penalty cost or, more generally, the recourse approach is based on a *quantitative risk concept*, whereas chance constrains are defined in terms of a *qualitative risk concept*. Indeed, the expected penalty costs as defined above measure the (expected) *amount* of surplus and/or shortage. An (individual) chance constraint on the other hand, measures the probability of having *any* shortage, irrespective of its size.

Whether it is preferable to use a qualitative risk measure depends on the application, of course. Chance constraints have been used successfully in a wide range of applications (see (references in) [16] and [20]), not in the last place because this risk model is very appealing to practitioners. However, it is also clear that there are applications for which quantitative risk constraints are more appropriate.

A second motivation for considering alternatives for chance constraints comes from their mathematical properties. It is well known that chance constrained problems are *non-convex* in general; they are (known to be) convex only if certain rather strong conditions on the distribution of the underlying random vector ω are satisfied. In particular, these conditions are not satisfied if ω follows a discrete distribution (see e.g. [18, 3]). This non-convexity causes great computational difficulties, the more so if chance constraints are used as one-stage risk constraints in multistage recourse models, which necessarily are modeled using discrete distributions. For example, such models have been proposed for financial applications [4, 5, 6].

In Section 2 of this paper we discuss properties of *integrated chance constraints* (ICC), which were proposed by Klein Haneveld [12] as a quantitative alternative for traditional chance constraints. In particular, we will present *reduced forms* for ICC with discretely distributed random vector ω. Subsequently, based on these reduced forms, we propose an algorithm for solving such ICC models in Section 3.

In Section 4 we discuss the relation between ICC and other model types, including the conditional value-at-risk concept in financial models, which has received a lot of (research) attention recently (see e.g. [1, 15, 17, 19]).

As a spin-off of our results for ICC models, we also present a special purpose algorithm for simple recourse models with discretely distributed random technology matrix T in Section 5.

Initial numerical results will be reported for both algorithms.

2 Integrated chance constraints

Integrated chance constraints (ICC) are defined in [12] as

$$\mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] \leq \beta_i, \qquad \beta_i \geq 0, \tag{2}$$

corresponding to individual chance constraints, and

$$\mathbb{E}_\omega \left[\max_{i\in I} \eta_i(x,\omega)^-\right] \leq \beta, \qquad \beta \geq 0, \tag{3}$$

as an alternative for a joint chance constraint. The risk aversion parameters β_i, $i \in I$, and β represent the largest acceptable expected shortfall, to be specified a priori by the decision maker. The name ICC refers to the fact that

$$\mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] = \int_{-\infty}^{0} \Pr\{\eta_i(x,\omega) < t\}\, dt,$$

see also Section 4.1.

The definitions (2) and (3) clearly show the quantitative nature of ICC. In contrast, the qualitative nature of traditional chance constraints is also immediate from their equivalent representations

$$\mathbb{E}_\omega \left[\operatorname{sgn}\left(\eta_i(x,\omega)^-\right)\right] \leq \alpha_i, \tag{4}$$

for an individual chance constraint, and

$$\mathbb{E}_\omega \left[\operatorname{sgn}\left(\max_{i\in I} \eta_i(x,\omega)^-\right)\right] \leq \alpha, \tag{5}$$

for a joint chance constraint, where $\operatorname{sgn}(z) = 1$ if $z > 0$ and $\operatorname{sgn}(z) = 0$ otherwise.

The differences in mathematical properties of ICC and chance constraints (CC) follow mostly from differences in their respective underlying loss functions. Indeed, the ICC loss function $(z)^-$ is convex, continuous, and strictly decreasing on $(-\infty, 0)$, whereas the CC loss function $\operatorname{sgn}\left((z)^-\right)$ is non-convex, discontinuous, and constant on $(-\infty, 0)$. The nice properties of the ICC loss function carry over to the corresponding feasible sets

$$C_1(\beta_i) := \left\{x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] \leq \beta_i\right\}, \qquad \beta_i \geq 0,$$

and

$$D_1(\beta) := \left\{x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\max_{i\in I} \eta_i(x,\omega)^-\right] \leq \beta\right\}, \qquad \beta \geq 0,$$

which are closed and convex *for any distribution of* ω, and increase smoothly with β_i (β) as the risk parameter increases from $\bar{\beta}$ to $+\infty$, where $\bar{\beta}$ is such that the set is non-empty

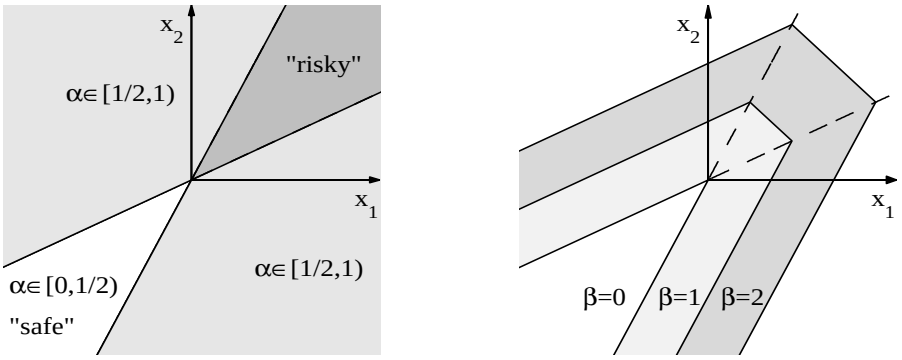

Figure 1: The feasibility sets $C_0(\alpha)$ (left) and $C_1(\beta)$ (right) of Example 2.1.

for $\beta \in [\bar{\beta}, +\infty)$ (see [12] for details). On the other hand, the unfavorable properties of the CC loss function are generally inherited by its expectation, so that the corresponding feasibility sets

$$C_0(\alpha_i) := \left\{ x \in \mathbb{R}^n : \Pr\{\eta_i(x,\omega)^- > 0\} \leq \alpha_i \right\}, \qquad \alpha_i \in [0,1],$$

and

$$D_0(\alpha) := \left\{ x \in \mathbb{R}^n : \Pr\{\eta_i(x,\omega)^- > 0,\ i \in I\} \leq \alpha \right\}, \qquad \alpha \in [0,1],$$

are convex only in exceptional cases, and may exhibit sudden jumps in increasing with the risk parameters α_i and α, respectively.

Example 2.1 Consider the goal constraint $\omega_1 x_1 + \omega_2 x_2 \geq 0$, so that risk is associated with shortages $\eta(x,\omega)^- = (\omega_1 x_1 + \omega_2 x_2)^-$, and assume that ω is discretely distributed with $\Pr\{(\omega_1,\omega_2) = (-2,1)\} = \Pr\{(\omega_1,\omega_2) = (1,-2)\} = 1/2$. (That is, $T(\omega) = (\omega_1\ \omega_2)$ and $h(\omega) = 0$; since $m = 1$ we dropped the index i.)

Then the feasible sets defined by the chance constraint $\Pr\{\eta(x,\omega)^- > 0\} \leq \alpha$, $\alpha \in [0,1)$, are given by

$$C_0(\alpha) = \begin{cases} \left\{ x \in \mathbb{R}^2 : 2x_1 - x_2 \leq 0,\ 2x_2 - x_1 \leq 0 \right\}, & 0 \leq \alpha < 1/2, \\ \left\{ x \in \mathbb{R}^2 : 2x_1 - x_2 \leq 0 \right\} \cup \left\{ x \in \mathbb{R}^2 : 2x_2 - x_1 \leq 0 \right\}, & 1/2 \leq \alpha < 1, \end{cases}$$

and $C_0(1) = \mathbb{R}^2$ of course.

On the other hand, the feasible sets defined by the ICC $\mathbb{E}_\omega\left[\eta(x,\omega)^-\right] \leq \beta$, $\beta \geq 0$, are given by

$$C_1(\beta) = \left\{ x \in \mathbb{R}^2 : \begin{array}{l} 2x_1 - x_2 \leq 2\beta, \\ 2x_2 - x_1 \leq 2\beta, \\ x_1 + x_2 \leq 2\beta \end{array} \right\}, \quad 0 \leq \beta \leq \infty,$$

see Section 2.1.

Both sets C_0 and C_1 are depicted in Figure 1. ◁

Example 2.1 clearly demonstrates that the ICC feasible set C_1 may behave much nicer than the corresponding CC feasible set C_0. Moreover, it also illustrates the following modeling issue. Suppose that the deterministic constraints, summarized as $x \in X$, are such that $X \subset R$, where $R := \{x \in \mathbb{R}^n : \Pr\{\eta(x,\omega)^- > 0\} = 1\}$ is the *extremely risky set*. The interpretation of this situation is clear, and it may well occur in practice: for any $x \in X$ a shortage is unavoidable. Then the optimization problem defined with the chance constraint is infeasible for all $\alpha \in [0,1)$, whereas the ICC alternative may be feasible already for

small values of the risk parameter β. In this case, the CC model does not differentiate at all between solutions in X, whereas the ICC model still allows to find an optimal solution $\bar{x}$ (with $\bar{x} \in R$, but the expected shortage associated to $\bar{x}$ small enough).

In the rest of this paper we restrict the analysis to the case that ω is discretely distributed. As already stated in the introduction, this is the setting in which traditional chance constraints give rise to non-convex feasible sets in general, but which is very common in applications. Indeed, such a discrete model of randomness is necessarily used in the context of multi-stage recourse models.

Our final result is an efficient algorithm to solve ICC models with finite discrete underlying distributions. This algorithm is based on the *reduced forms* of the sets C_1 and D_1, that is, representations that make these ICC models amenable to (standard) optimization algorithms. In the following sections we derive closed forms for the feasible sets defined by (two versions of) individual ICC as well as joint ICC constraints. See [12] for closed forms for the case that ω is normally distributed.

2.1 Individual integrated chance constraints

We first consider the individual integrated chance constraint

$$\mathbb{E}_\omega \left[\eta(x,\omega)^-\right] \leq \beta, \qquad \beta \in \mathbb{R}_+,$$

where, to avoid notational burden, we assume for the time being that $\eta(x,\omega) = T(\omega)x - h(\omega) \in \mathbb{R}$ (i.e., $m = 1$) for $x \in \mathbb{R}^n$ and ω a random vector. This individual ICC represents the wish of the decision maker to avoid large infeasibilities with respect to the single random constraint $T(\omega)x \geq h(\omega)$.

As before, we define

$$C_1(\beta) := \left\{x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\eta(x,\omega)^-\right] \leq \beta\right\},$$

the feasible set corresponding to this constraint. Since $\eta(x,\omega)^-$ is a convex function of x for each fixed ω, the set $C_1(\beta)$ is defined as the lower level set of a convex function; hence, the set $C_1(\beta)$ is convex for all distributions of ω and for all $\beta \in \mathbb{R}$ (but empty if $\beta < 0$).

Lemma 2.1 *Assume that ω is a* discrete *random vector, with* $\Pr\{\omega = \omega^s\} = p^s$, $s \in S$, *and let $(T^s, h^s) = (T(\omega^s), h(\omega^s))$ for $s \in S$. Then, for $\beta \geq 0$,*

$$C_1(\beta) = \bigcap_{K \subset S} \left\{x \in \mathbb{R}^n : \sum_{k \in K} p^k \left(h^k - T^k x\right) \leq \beta\right\}. \tag{6}$$

If S is a finite set, then $C_1(\beta)$ is a polyhedral set defined by $2^{|S|} - 1$ linear constraints.

PROOF. Since ω is discretely distributed, we have

$$\begin{aligned}
\mathbb{E}_\omega \left[\eta(x,\omega)^-\right] &= \sum_{s \in S} p^s \max\{0, -\eta(x,\omega^s)\} \\
&= \sum_{s \in S} \max\{0, -p^s \eta(x,\omega^s)\} \\
&= \sum_{s \in S} \left(-p^s \eta(x,\omega^s)\right)^+ \\
&= \max_{K \subset S} \sum_{k \in K} -p^k \eta(x,\omega^k), \qquad (7)
\end{aligned}$$

where the last equality follows from the observation that $\max_{K \subset S} \sum_{k \in K} a_k$, $a_k \in \mathbb{R}$, is attained for the subset $\{k \in S : a_k > 0\}$. Moreover, for $m_i \in \mathbb{R}$, $i \in N$, with N an arbitrary

index set, it holds that $\sup_{i\in N} m_i \leq M$ if and only if $m_i \leq M$ for all $i \in N$, so that

$$\begin{aligned} C_1(\beta) &= \bigcap_{K\subset S} \left\{ x \in \mathbb{R}^n : -\sum_{k\in K} p^k \eta(x, \omega^k) \leq \beta \right\} \\ &= \bigcap_{K\subset S} \left\{ x \in \mathbb{R}^n : \sum_{k\in K} p^k \left(h^k - T^k x \right) \leq \beta \right\}. \end{aligned}$$

If S is finite then there are $2^{|S|} - 1$ non-empty subsets of S, so that (6) describes the convex set $C_1(\beta)$ using finitely many linear constraints. That is, $C_1(\beta)$ is a polyhedral set in this case. □

Remark 2.1 For $K = S$ in (6), we obtain $\bar{T}x \geq \bar{h} - \beta$, where $(\bar{T}, \bar{h}) = \mathbb{E}_\omega [(T(\omega), h(\omega))]$. By our assumption that $T(\omega)$ and $h(\omega)$ depend linearly on ω, this is equivalent to $\eta(x, \mathbb{E}_\omega [\omega])^- \leq \beta$. This is an obvious necessary condition for $x \in C_1(\beta)$, since $\eta(x, \omega)^- \geq -\eta(x, \omega)$ for all ω so that $\mathbb{E}_\omega \left[\eta(x, \omega)^-\right] \geq -\eta(x, \mathbb{E}_\omega [\omega])$.

Also, the subset $K = \emptyset$ gives the trivial constraint $0 \leq \beta$, which stresses the obvious fact that $C_1(\beta) = \emptyset$ for all $\beta < 0$.

Remark 2.2 Lemma 2.1 can be generalized to a result for arbitrary distributions, as follows: for $\beta \geq 0$,

$$C_1(\beta) = \bigcap_{B\in\mathcal{B}} \left\{ x \in \mathbb{R}^n : \mathbb{E}_\omega [-\eta(x, \omega) \cdot I_B(\omega)] \leq \beta \right\}, \tag{8}$$

where $\mathcal{B}$ is the collection of all Borel sets in $\mathbb{R}$, and the indicator function $I_B(\omega) = 1$ if and only if $\omega \in B$.

Because we will only consider discrete distributions in the rest of this paper, we omit the rather involved proof of (8).

We conclude from Lemma 2.1 that problems with individual ICC constraints and finite distributions can be represented as LP problems. In fact, by using discrete approximations of arbitrary (continuous) distributions, Lemma 2.1 implies that any ICC problem can at least be approximated by an LP problem. Note that such a discrete approximation does not work well for traditional CC problems, since in that case the approximating problem is non-convex in general.

However, for a realistic number of realizations S the number of constraints in the LP formulation is huge, so that it is not efficient to solve these problems by brute force LP techniques. Instead, in Section 3 we propose an efficient algorithm which is based on the reduced form of the set C_1 presented above.

2.2 Individual integrated chance constraints: second type

In addition to the nice mathematical properties of ICC constraints, we argued that they also have nice modeling properties. From the latter point of view, it may be considered a disadvantage that the risk parameters of ICC constraints are not scale-free. In this section we review an alternative individual ICC constraint as proposed in [12], which does not suffer from this shortcoming. Again, we derive a reduced form for the induced feasible set.

Instead of specifying the maximum acceptable risk as a fixed number β, it can be chosen dependent on the distribution of $\eta(x, \omega)$. Since $\mathbb{E}_\omega [|\eta(x, \omega)|]$ is a natural upper bound for $\mathbb{E}_\omega \left[\eta(x, \omega)^-\right]$, we define, for $\alpha \in [0, 1]$ to be specified in advance, the ICC constraint

$$\mathbb{E}_\omega \left[\eta(x, \omega)^-\right] \leq \alpha \cdot \mathbb{E}_\omega [|\eta(x, \omega)|],$$

with corresponding feasible set

$$\begin{aligned} C_2(\alpha) &:= \left\{ x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\eta(x,\omega)^- \right] \leq \alpha \cdot \mathbb{E}_\omega \left[|\eta(x,\omega)| \right] \right\} \\ &= \left\{ x \in \mathbb{R}^n : (1-2\alpha)\mathbb{E}_\omega \left[\eta(x,\omega)^- \right] \leq \alpha \cdot \mathbb{E}_\omega \left[\eta(x,\omega) \right] \right\}, \end{aligned} \tag{9}$$

where the equality follows using $|t| = (t)^- + (t)^+ = (t)^- + (t + (t)^-)$, $t \in \mathbb{R}$.

Lemma 2.2 *Assume that ω is a* discrete *random vector, with* $\Pr\{\omega = \omega^s\} = p^s$, $s \in S$.

(i) If $\alpha \in [0, 1/2]$ then

$$C_2(\alpha) = \bigcap_{K \subset S} \left\{ x \in \mathbb{R}^n : (1-2\alpha) \sum_{k \in K} p^k \left(h^k - T^k x \right) \leq \alpha \left(\bar{T} x - \bar{h} \right) \right\}$$

where $(T^s, h^s) = (T(\omega^s), h(\omega^s))$, $s \in S$, and $(\bar{T}, \bar{h}) = \mathbb{E}_\omega [(T(\omega), h(\omega))]$.

If S is a finite set, then $C_2(\alpha)$ is a polyhedral convex set defined by $2^{|S|} - 1$ linear constraints.

(ii) If $\alpha > 1/2$ then

$$C_2(\alpha) = \bigcup_{K \subset S} \left\{ x \in \mathbb{R}^n : (1-2\alpha) \sum_{k \in K} p^k \left(h^k - T^k x \right) \leq \alpha \left(\bar{T} x - \bar{h} \right) \right\}.$$

In this case $C_2(\alpha)$ is the union *of convex sets (polyhedra if S is finite), which is non-convex in general.*

PROOF. (i) If $\alpha < 1/2$ (so that $1 - 2\alpha > 0$), the same approach as in the proof of Lemma 2.1 leads to the result. For $\alpha = 1/2$ the result is trivial.

(ii) If $\alpha > 1/2$ we find that

$$C_2(\alpha) = \left\{ x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\eta(x,\omega)^- \right] \geq \frac{\alpha}{1-2\alpha} \cdot \mathbb{E}_\omega \left[\eta(x,\omega) \right] \right\}.$$

Applying (7), and using that $\max_{i \in N} m_i \geq M$ if and only if $\exists i \in N : m_i \geq M$ (with $m_i \in \mathbb{R}$, $i \in N$; N finite), we obtain

$$\begin{aligned} C_2(\alpha) &= \bigcup_{K \subset S} \left\{ x \in \mathbb{R}^n : \sum_{k \in K} -p^k \eta(x, \omega^k) \geq \frac{\alpha}{1-2\alpha} \mathbb{E}_\omega \left[\eta(x,\omega) \right] \right\} \\ &= \bigcup_{K \subset S} \left\{ x \in \mathbb{R}^n : (1-2\alpha) \sum_{k \in K} p^k \left(h^k - T^k x \right) \leq \alpha \left(\bar{T} x - \bar{h} \right) \right\}. \end{aligned}$$

□

Remark 2.3 By equation (9), $C_2(1/2) = \{x \in \mathbb{R}^n : \bar{T} x \geq \bar{h}\}$. Thus, the naive modeling choice to ignore uncertainty by replacing the random coefficients in the constraint by their expected values, corresponds to an ICC constraint with risk aversion parameter $\alpha = 1/2$.

In practice, risk aversion is modeled by choosing $\alpha \ll 1/2$. As can be seen from Lemma 2.2, the set $C_2(\alpha)$ is convex polyhedral in this case, at least if the underlying distribution is finite. In Section 3 we will show that our algorithm can be adapted to handle this type of individual ICC constraints.

2.3 Joint integrated chance constraints

We now turn to joint integrated chance constraints, which are defined in [12] as

$$\mathbb{E}_\omega \left[\max_{i \in I} \eta_i(x, \omega)^- \right] \leq \beta, \qquad \beta \geq 0, \tag{10}$$

with $\eta_i(x, \omega) := T_i(\omega)x - h_i(\omega)$, $i \in I := \{1, \ldots, m\}$. From a formal point of view, this definition is analogous to the definition of individual ICC (first type): in both cases the ICC variant is obtained by dropping the $\operatorname{sgn}(\cdot)$ operator in the alternative definitions (4) and (5) of the CC variants. See [12] for joint variants of individual ICC of the second type, and generalizations of ICC for which no traditional CC companion exists.

The feasibility set induced by (10) is

$$D(\beta) := \left\{ x \in \mathbb{R}^n : \mathbb{E}_\omega \left[\max_{i \in I} \eta_i(x, \omega)^- \right] \leq \beta \right\},$$

which is obviously convex for all distributions of ω and for all values of the risk parameter β.

Lemma 2.3 *Assume that ω is a* discrete *random vector, with* $\Pr\{\omega = \omega^s\} = p^s$, $s \in S$, *with S a finite set. Denote $(T_i^s, h_i^s) := (T_i(\omega^s), h_i(\omega^s))$, $i \in I$, $s \in S$. Then*

$$D(\beta) = \bigcap_{K \subset S} \bigcap_{l \in I^K} \left\{ x \in \mathbb{R}^n : \sum_{k \in K} -p^k \left(T_{l_k}^k x - h_{l_k}^k \right) \leq \beta \right\},$$

where $I^K := \{l = (l_k, k \in K) : l_k \in I \text{ for all } k \in K\}$. That is, for any subset K of realizations, select for every realization $k \in K$ a row l_k of (T^k, h^k).

Since S is a finite set, the set $D(\beta)$ is polyhedral; it is defined by $(m+1)^{|S|} - 1$ linear constraints.

PROOF. Since ω is a discrete random vector, we have

$$\begin{aligned} \mathbb{E}_\omega \left[\max_{i \in I} \eta_i(x, \omega)^- \right] &= \sum_{s \in S} \max \left\{ 0, \max_{i \in I} -p^s \eta_i(x, \omega^s) \right\} \\ &= \sum_{s \in S} \max_{i \in I_0} y_i^s, \end{aligned} \tag{11}$$

where $I_0 := I \cup \{0\}$, $y_0^s := 0$, and $y_i^s := -p^s \eta_i(x, \omega^s)$, $i \in I$.

Define $\mathcal{F}$ as the family of all mappings $f : S \mapsto I_0$. Then from (11) we get

$$\mathbb{E}_\omega \left[\max_{i \in I} \eta_i(x, \omega)^- \right] = \max_{f \in \mathcal{F}} \sum_{s \in S} y_{f(s)}^s,$$

so that

$$D(\beta) = \left\{ x \in \mathbb{R}^n : \sum_{s \in S} y_{f(s)}^s \leq \beta \quad \forall f \in \mathcal{F} \right\}.$$

Since $|\mathcal{F}| = |I_0|^{|S|} = (m+1)^{|S|}$, this is a polyhedral representation of $D(\beta)$ in terms of $(m+1)^{|S|} - 1$ linear inequalities. (The case that $f(s) = 0$ $\forall s$ indicates that β should be nonnegative, but this does not provide a constraint on x.)

By substitution of the values of y_i^s, and separating $i = 0$ from $i \geq 1$, we get the claimed formula for the reduced form of $D(\beta)$. □

In the next section we show how this characterization of the set $D(\beta)$ can be used to construct an algorithm for solving problems with joint integrated chance constraints.

3 Algorithm for ICC

For each of the three types of integrated chance constraints considered above, the reduced form is an explicit description of the feasible set which is convex polyhedral (at least if $|S|$ is finite). However, in all cases the number of linear constraints is already huge for small problem instances, so that it is impractical or even impossible to solve the corresponding problems by a standard LP algorithm.

At the same time, it is easy to see that the following system of linear constraints is equivalent to an individual ICC (first type) with finite discrete distribution:

$$\begin{array}{lll} T^s x + y^s & \geq h^s, & s \in S \\ \sum\limits_{s \in S} p^s y^s & \leq \beta & \\ y^s \geq 0, & & s \in S. \end{array} \tag{12}$$

Hence, such an individual ICC can be represented in LP form at the cost of $|S| + 1$ constraints and $|S|$ additional variables. Similar representations can be constructed for other types of ICC, see Section 4.1. Given the availability of very powerful LP software, the resulting LP problem can be solved efficiently if $|S|$ is not too large. Indeed, for small problem instances, this approach outperforms our special purpose algorithm that will be presented next. However, already for instances of moderate size, our algorithm is (much) more efficient.

3.1 Individual ICC (first type)

The main idea of our special purpose algorithm is that it uses only a partial description of the feasible set. Below we propose such an algorithm for the case of an individual chance constraint of the first type. Next, we indicate how to adapt the algorithm for the other two cases.

Remark 3.1 We assume that the problem

$$\begin{array}{ll} \min\limits_x & cx \\ \text{s.t.} & x \in X \cap C^0, \end{array}$$

with $C^0 := \{x \in \mathbb{R}^n : \bar{T}x \geq \bar{h} - \beta\}$ (see Remark 2.1), is bounded. If not, this can be forced by including suitable simple bounds in the definition of the set X.

Algorithm 3.1 *Define the* current problem *CP at iteration t as*

$$\begin{array}{ll} \min\limits_x & cx \\ \text{s.t.} & x \in X \\ & x \in C^t := \left\{x \in \mathbb{R}^n : d_i x \leq e_i, \quad i = 0, \ldots, t\right\} \end{array}$$

where the feasibility cuts $d_i x \leq e_i$, $i = 0, \ldots, t$, *give a partial description of the feasible set $C_1(\beta)$. Set $t = 0$.*

Iteration t*:*

(i) Solve the LP problem CP, giving an optimal solution x^t. If CP is infeasible STOP*: the problem is infeasible.*

(ii) Compute $\mathbb{E}_\omega\left[\eta(x^t, \omega)^-\right] = \sum_{s \in S} p^s \eta(x^t, \omega^s)^-$, at the same time constructing the index set

$$K^t := \left\{s \in S : \eta(x^t, \omega^s)^- > 0\right\}.$$

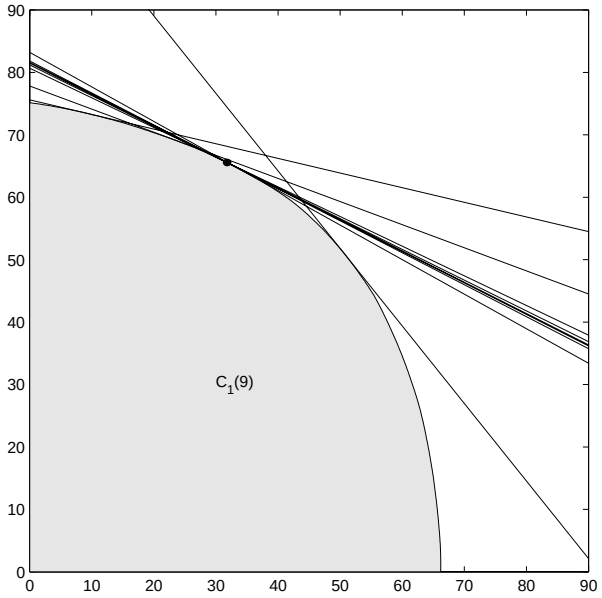

Figure 2: Illustration of Algorithm 3.1, see Example 3.1.

(iii) If $\mathbb{E}_\omega\left[\eta(x^t,\omega)^-\right] \le \beta$ STOP: *x^t is an optimal solution. Otherwise, construct a feasibility cut* $d_{t+1}x \le e_{t+1}$, *with*

$$d_{t+1} = -\sum_{k\in K^t} p^k T^k, \qquad e_{t+1} = \beta - \sum_{k\in K^t} p^k h^k,$$

and set

$$C^{t+1} = C^t \cap \left\{x \in \mathbb{R}^n : d_{t+1}x \le e_{t+1}\right\}.$$

Proceed with iteration $t+1$.

Theorem 3.1 *Consider the bounded optimization problem with individual ICC (first type)*

$$\begin{array}{ll}\min\limits_x & cx\\ \text{s.t.} & x \in X\\ & \mathbb{E}_\omega\left[\eta(x,\omega)^-\right] \le \beta,\end{array}$$

with β a given risk parameter. Assume that ω follows a finite discrete distribution. Then Algorithm 3.1 solves this problem in finitely many iterations.

PROOF. In each iteration, the algorithm either finds that the problem is infeasible, adds a new feasibility cut, or returns an optimal solution. The algorithm is finite since there are only finitely many feasibility cuts by Lemma 2.1. □

Example 3.1 Consider the ICC problem specified by $c = (-1\ -2)$, $X = [0,100]^2$, and the individual ICC constraint

$$\mathbb{E}_\omega\left[\left(\omega_1 x_1 + \omega_2 x_2 - \omega_3\right)^-\right] \le \beta,$$

where the 1000 realizations of $\left(T(\omega), h(\omega)\right) = \left(\omega_1, \omega_2, \omega_3\right)$ are sampled from the uniform distribution on $[-.5, .5]^2 \times [0,1]$, and $\beta = 9$.

The optimal solution $x^\star = (31.71, 65.58)$ is found by Algorithm 3.1 after 10 iterations, i.e., using only 9 feasibility cuts (out of the $2^{1000} \approx 10^{301}$ cuts constituting the reduced form of $C_1(\beta)$ according to Lemma 2.1). See Figure 2. ◁

3.2 ICC variants

Algorithm 3.1 also applies to individual ICC of the second type with $\alpha < 1/2$, that is, such that $C_2(\alpha)$ is convex. In this case, the set $C^0 = \{x \in \mathbb{R}^n : \bar{T}x \geq \bar{h}\}$, see Remark 2.3. The only further changes are in step (iii) of each iteration: check if $(1-2\alpha)\mathbb{E}_\omega\left[\eta(x^t,\omega)^-\right] \leq \alpha \cdot \left(\bar{T}x^t - \bar{h}\right)$ (then the current solution x^t is optimal); otherwise, with K^t the index set

$$K^t = \left\{s \in S : (1-2\alpha)\eta(x^t,\omega^s)^- \leq \alpha \cdot \left(\bar{T}x^t - \bar{h}\right)\right\},$$

generate a feasibility cut with parameters

$$d_{t+1} = -\alpha\bar{T} - (1-2\alpha)\sum_{k\in K^t} p^k T^k, \qquad e_{t+1} = -\alpha\bar{h} - (1-2\alpha)\sum_{k\in K^t} p^k h^k,$$

see Lemma 2.2.

With minor modifications, Algorithm 3.1 can also be used to solve problems with a joint ICC. The optimality check changes in the obvious way. In iteration t, the index set K^t becomes

$$K^t = \left\{(k,l) \in S \times I : \eta_l^k(x^t,\omega^k) = \max_{i\in I} \eta_i^k(x^t,\omega^k)^-, \eta_l^k(x^t,\omega^k) > 0\right\},$$

and the parameters of the feasibility cut at iteration t are given by

$$d_{t+1} = -\sum_{(k,l)\in K^t} p^k T_l^k, \qquad e_{t+1} = \beta - \sum_{(k,l)\in K^t} p^k h_l^k,$$

see Lemma 2.3.

3.3 Numerical results

Algorithm 3.1 is implemented in Matlab [8], using MOSEK [7] to solve the current LP problems.

To test the algorithm, problem instances were randomly generated, as follows. For a given dimension n of the decision variables and number of realizations $|S|$

- A cost vector c is sampled from the uniform distribution on $[-1, 0]^n$; thus, effectively $-cx$ will be maximized.
- The set X equals $[0, 10000]^n$.
- d% of the components of the n-vector $T(\omega)$ are random; the remaining $(100-d)$% are deterministic entries. The fixed elements are obtained as a sample from the uniform distribution on $[-1, 1]$, whereas random elements of the realizations T^s, $s \in S$, are sampled from uniform distributions on $[-1, 0]$ or $[0, 1]$ ($d/2$% each way).
- Realizations h^s, $s \in S$, are sampled from the uniform distribution on $[0, 1]$.
- The probabilities p^s, $s \in S$, are also drawn from the uniform distribution on $[0, 1]$ (and then normalized).
- The risk parameter β equals 12.345 for all instances.

The tables below show computational results (on a Pentium III 450 Mhz 384 MB, Win NT 4.0) of Algorithm 3.1 for these maximization problems with an individual ICC of the first type. The parameter d, specifying the percentage of random elements of $T(\omega)$, equals 50. In all tables, the results are average values for 10 instances of the given dimensions: n is the the dimension of x and $|S|$ is the number of realizations of $(T(\omega), h(\omega))$.

For each group of random instances, Table 1 shows the average number of cuts used by Algorithm 3.1 to find an optimal solution.

n	$\|S\| = 10$	100	1000	5000	10000
2	1.9	4.0	5.8	6.0	6.3
5	5.2	16.0	27.6	32.3	36.0
10	3.0	19.2	44.5	53.4	62.6
15	3.5	18.8	81.3	77.1	130.7
20	4.0	22.3	79.3	141.6	175.6
25	4.3	27.6	100.1	199.8	236.9
50	3.7	19.1	55.2	178.4	288.0
100	4.0	14.5	61.1	110.2	203.2
200	3.1	18.3	40.9	55.6	79.4
300	4.1	12.9	35.4	71.9	65.4

Table 1: Average number of feasibility cuts used by Algorithm 3.1.

We see that the algorithm is very efficient in the sense that it uses only a small fraction of the $2^{|S|} - 1$ cuts which describe the set $C_1(\beta)$ (see Lemma 2.1). For each fixed n the average number of cuts increases with $|S|$ (with a few exceptions), which agrees with our intuition. However, such monotonicity is not observed for fixed $|S|$: most cuts are used for instances with n equal to 25 or 50. At this time, we do not have an explanation for this behavior.

Table 2 shows the average CPU times used by Algorithm 3.1, which was implemented using the warm-start capability of the primal simplex solver of MOSEK. Since the current version of MOSEK does not provide a call back to include an additional constraint, the solver had to be restarted for each iteration of Algorithm 3.1. To correct for this overhead, which we accurately estimated to be 0.1 CPU seconds for each restart, we subtracted $0.1 \times N(n, |S|)$ from the actual CPU times, where $N(n, |S|)$ is the average number of cuts (i.e., the number of restarts) as reported in Table 1. Thus, for example, the actual average CPU time for instances with $n = 2$ and $|S| = 100$ was $0.26 + 0.1 \times 4.0 = 0.66$ seconds.

The results from Table 2 show that Algorithm 3.1 solves these ICC problems fast enough to be practical, even for relatively large instances.

n	$\|S\| = 10$	100	1000	5000	10000
2	0.19	0.26	0.31	0.38	0.46
5	0.30	0.65	1.03	1.47	1.99
10	0.22	0.74	1.65	2.69	4.12
15	0.23	0.72	3.11	4.36	10.19
20	0.26	0.83	3.25	9.02	16.07
25	0.27	1.03	4.25	14.70	25.51
50	0.26	0.79	2.92	20.03	53.92
100	0.26	0.69	4.50	20.06	66.16
200	0.27	1.04	4.42	17.69	46.02
300	0.32	0.87	5.19	33.26	55.76

Table 2: Average CPU times (in seconds) for Algorithm 3.1.

As mentioned in the introduction of this section, an individual ICC can be represented by the system of linear inequalities (12). The resulting LP problem, having $|S| + 1$ additional constraints and $|S|$ additional variables, can then be solved by any standard LP algorithm. Table 3 shows the average relative CPU times for Algorithm 3.1 compared to this direct LP approach. The latter LP problems were solved using the interior point solver of MOSEK, which turned out to be faster than the simplex solver.

n	$\|S\| = 10$	100	1000	5000	10000
2	1.266	1.825	1.164	0.330	0.158
5	2.203	3.823	1.348	0.225	0.120
10	1.610	4.089	1.035	0.102	0.018
15	1.483	3.772	0.736	0.059	0.031
20	1.894	3.444	0.271	0.071	0.033
25	1.954	4.007	0.357	0.084	0.065
50	1.800	2.510	0.173	0.059	0.063
100	1.564	1.474	0.079	0.050	0.058
200	1.526	1.194	0.037	0.020	0.020
300	1.530	0.656	0.032	0.019	0.017

Table 3: Average relative CPU times for Algorithm 3.1.

As expected, our algorithm can not beat the LP solver for small problem instances. However, Table 3 shows clearly that Algorithm 3.1 is much faster on larger instances. In practice, the difference for small instances may not be very important (e.g., on average 19.2 versus 4.8 seconds for instances with $n = 10$ and $|S| = 100$). In contrast, for instances with $n = 300$ and $|S| = 10000$ the respective average CPU times are approximately 1 minute versus 1 hour.

We also tested Algorithm 3.1 on similar sets of problem instances for other values of the density parameter d. We found that the relative advantage of Algorithm 3.1, as reflected in Table 3, decreases with d. For example, for $d = 100$, our algorithm is faster only for instances with $n \geq 200$ and $|S| \geq 5000$. On the other hand, with the (more realistic) parameter choice $d = 10$, the average relative CPU time ranges from 1.95 to 0.005, with values larger than 1 only for a few small instances.

4 Relation to other models

In this section we discuss the relation between ICC models and several other well-known models to cope with uncertainty. First we take a second look at traditional CC models. Next, we consider *Conditional Surplus-at Risk* (CSaR) models, which are increasingly popular in financial applications. Finally, we discuss the relation to recourse models, in particular to *simple recourse* (SR) models. The close mathematical relation between ICC and SR models is exploited in the next section, where we will present a special purpose algorithm for the latter model type.

4.1 ICC and chance constraints

As discussed in Sections 1 and 2, traditional CC models are of a qualitative nature: they restrict the probability of shortfall with respect to a given level l (usually, $l = 0$), irrespective of its size. However, in many applications a large shortfall is disliked more than a small one. Within the context of CC models this can be modeled by using several CC constraints, each corresponding to a different shortfall level; of course, the respective risk parameters should be chosen to reflect larger dislike for larger shortfalls. For example, with L a (finite) subset of $\mathbb{R}_-$, we may define a number of individual chance constraints

$$\Pr\{\eta_i(x, \omega) < l\} \leq \alpha(l), \quad l \in L,$$

with $\alpha(l)$ increasing with l. If $L = \mathbb{R}_-$, that is, if we include a CC constraint for every shortfall level, we obtain infinitely many CC constraints, which may be handled by

aggregating them, giving

$$\mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] = \int_{-\infty}^{0} \Pr\{\eta_i(x,\omega) < l\} dl \leq \int_{-\infty}^{0} \alpha(l) dl =: \beta, \tag{13}$$

which we recognize as an individual ICC constraint. Formula (13) explains why Klein Haneveld [12] proposed the name *integrated chance constraint*.

Remark 4.1 The equality in (13) follows from the well-known formula for the *expected shortfall function* of the random yield ξ,

$$\mathbb{E}_\xi \left[(\xi - z)^-\right] = \int_{-\infty}^{z} \Pr\{\xi < t\} dt, \quad z \in \mathbb{R},$$

evaluated at $z = 0$.

Next, consider an individual CC constraint with discretely distributed random parameters, specified by $\Pr\{(T(\omega), h(\omega)) = (T^s, h^s)\} = p^s$, $s \in S$. If S is finite, then this CC can be represented in mixed-integer LP format as

$$\begin{array}{lll} T^s x + M\delta^s & \geq h^s, & s \in S \\ \sum\limits_{s\in S} p^s \delta^s & \leq \alpha & \\ \delta^s \in \{0,1\}, & & s \in S, \end{array}$$

where M is a sufficiently large number. If we relax the integrality constraints, and substitute $y^s := M\delta^s$ and $\beta := M\alpha$, we obtain

$$\begin{array}{lll} T^s x + y^s & \geq h^s, & s \in S \\ \sum\limits_{s\in S} p^s y^s & \leq \beta & \\ y^s \in [0, M], & & s \in S, \end{array}$$

which is equivalent to the LP representation (12) of the corresponding individual ICC of the first type. Analogously, a joint CC constraint with finite discrete distribution can be written as

$$\begin{array}{lll} T_i^s x + M\delta_i^s & \geq h_i^s, & s \in S,\ i \in I \\ \Delta^s & \geq \delta_i^s, & s \in S,\ i \in I \\ \sum\limits_{s\in S} p^s \Delta^s & \leq \alpha & \\ \delta_i^s \in \{0,1\}, & & s \in S,\ i \in I \\ \Delta^s \in \{0,1\} & & s \in S, \end{array}$$

which on relaxation yields an LP representation of the corresponding joint ICC constraint:

$$\begin{array}{lll} T_i^s x + y_i^s & \geq h_i^s, & s \in S,\ i \in I \\ z^s & \geq y_i^s, & s \in S,\ i \in I \\ \sum\limits_{s\in S} p^s z^s & \leq \beta & \\ y_i^s \in [0, M], & & s \in S,\ i \in I \\ z^s \geq 0, & & s \in S, \end{array}$$

as can be verified easily.

ICC constraints can therefore also be seen as relaxations of traditional CC constraints, at least if the underlying distribution is discrete. In this sense, ICC constraints are natural candidates to replace CC constraints, in particular in the context of multistage recourse

models which necessarily use discrete distributions. In [4] such a multistage model with CC constraints is proposed to model asset-liability management for pension funds, whereas [6] considers the same application using ICC constraints.

Moreover, this relation between CC and ICC constraints suggests that ICCs could be used to construct convex approximations of the (generally non-convex) feasible sets induced by CC constraints. Our initial results in this line of research are promising; this subject will be worked out in a subsequent paper.

4.2 ICC and CSaR

In this section we consider the relation between ICC constraints and *Conditional Surplus-at-Risk* (CSaR) constraints. CSaR is a variant of the Conditional Value-at-Risk concept, which is very popular, in particular in financial applications. Here we consider CSaR because it is defined in terms of surplus, similar to ICC which is based on shortage. We refer to [19] for an overview of recent results in this area.

Let the random variable ξ represent a yield, so that higher values are preferred. For the moment, we suppress the dependence of ξ on the decision variables x, e.g., $\xi = \eta(\omega, x)$. To avoid technicalities, we will assume that ξ has finite mean value μ and is continuously distributed, with density function f satisfying $f(t) > 0$ for all $t \in \mathbb{R}$, so that its distribution function F is invertible.

Definition 4.1 For $\gamma \in (0, 1)$, the *Conditional Surplus-at-Risk* at probability level γ, associated with the random variable ξ, is

$$c(\gamma) := \mathbb{E}_\xi \left[\xi \mid \xi \leq v(\gamma)\right],$$

where $v(\gamma)$ is the *Surplus-at-Risk* at level γ,

$$v(\gamma) := \max_{t \in \mathbb{R}} \{t : \Pr\{\xi > t\} \geq \gamma\}.$$

Thus, Surplus-at-Risk (SaR) at level γ is defined as the $(1-\gamma)$-quantile of the distribution of ξ, and CSaR is the conditional expectation of the tail of the distribution up to SaR. It follows from our assumption on the distribution of ξ that $v(\gamma) = F^{-1}(1-\gamma)$, $\gamma \in (0, 1)$.

We will show that the CSaR constraint

$$c(\gamma, x) \geq c_0$$

is closely related to an individual ICC constraint of the first type. The notation $c(\gamma, x)$ reflects the dependence of the random surplus on the decision variables x; the parameter c_0 is the minimal required conditional surplus at risk.

First we show that SaR and CSaR can be found as the solution and the optimal value, respectively, of a convex optimization problem involving the expected shortage function $H(t) := \mathbb{E}_\xi \left[(\xi - t)^-\right]$, $t \in \mathbb{R}$. Properties of the convex function H are well-known from the study of simple recourse models.

Lemma 4.1 *(i) For $\gamma \in (0, 1)$, the SaR $v(\gamma)$ satisfies*

$$v(\gamma) = \operatorname*{argmax}_{t \in \mathbb{R}} \left(t - \frac{H(t)}{1-\gamma} \right).$$

(ii) For $\gamma \in (0, 1)$, the CSaR $c(\gamma)$ satisfies

$$c(\gamma) = \max_{t \in \mathbb{R}} \left(t - \frac{H(t)}{1-\gamma} \right),$$

so that

$$c(\gamma) = v(\gamma) - \frac{H(v(\gamma))}{1-\gamma}.$$

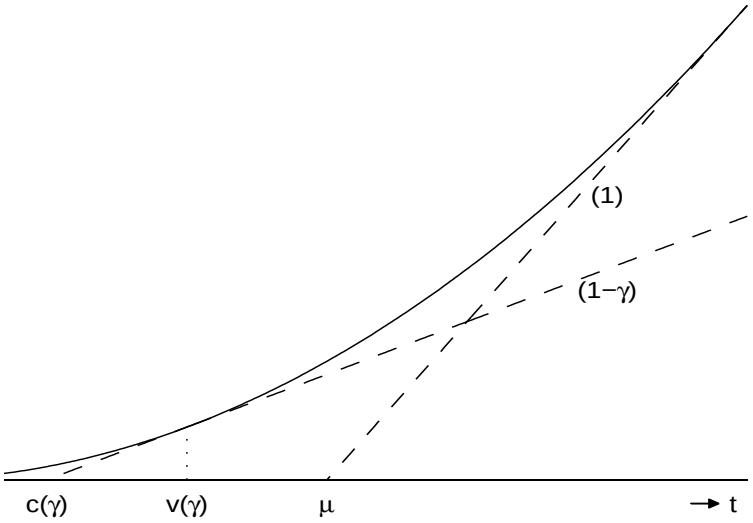

Figure 3: Illustration of Lemma 4.1: CSaR and SaR related to the function H (solid). CSaR equals the value of t where the tangent with slope $(1-\gamma)$ intersects the t-axis; SaR is the t-coordinate of the tangent point.

PROOF. (i) Because ξ is continuously distributed, the function H is differentiable with derivative F. The first part then follows from the observation that $H'(v(\gamma)) = F(v(\gamma)) = 1-\gamma$.

(ii) By Definition 4.1, $c(\gamma) = m(v(\gamma))$ with $m(t) := \mathbb{E}_\xi[\xi|\xi \le t]$, $t \in \mathbb{R}$. Since, for $t \in \mathbb{R}$,

$$\begin{aligned} H(t) &= \Pr\{\xi \le t\} \cdot \mathbb{E}_\xi[t-\xi|\xi \le t] + \Pr\{\xi > t\} \cdot 0 \\ &= F(t) \cdot \big(t - \mathbb{E}_\xi[\xi|\xi \le t]\big) \\ &= F(t) \cdot \big(t - m(t)\big), \end{aligned}$$

it follows that

$$m(t) = t - \frac{H(t)}{F(t)}, \qquad t \in \mathbb{R}.$$

Part (ii) now follows from (i) by substitution. □

See Figure 3 for an illustration of Lemma 4.1.

Remark 4.2 Similar results, formulated in terms of the convex conjugate of the function H, have been obtained independently in [14].

In the context of decision models it is natural to assume that the random surplus depends on the decision variables $x \in \mathbb{R}^n$. In agreement with previous notation, we assume that the surplus is $\eta(\omega, x) = T(\omega)x - h(\omega)$, where the n-vector $T(\omega)$ and the scalar $h(\omega)$ depend linearly on the underlying random vector ω.

By Lemma 4.1, the CSaR constraint $c(\gamma, x) \ge c_0$ is equivalent to

$$\mathbb{E}_\omega\Big[\big(\eta(\omega, x) - t\big)^-\Big] \le (1-\gamma)(t - c_0), \qquad t = v(\gamma). \tag{14}$$

If we choose to consider t as a parameter instead, with the interpretation of a threshold level to be met, (14) is just an individual ICC constraint of the first type. For any fixed

t, the aspiration parameter c_0 should not be set too high, since then no feasible solutions exist. This corresponds to the fact that $v(\gamma)$ is a natural upper bound for c_0 in the CSaR constraint, since $c(\gamma) \leq v(\gamma)$ by definition.

4.3 ICC and Simple Recourse

Just as in ICC models, in simple recourse (SR) models the risk measure is *mean shortage* (and/or its counterpart mean surplus). However, in SR models risk aversion is specified in terms of penalty costs for shortages rather than by prescribing the maximum acceptable risk. Clearly, there must be an intimate relation between both model types, expressed by Lagrange multipliers.

Consider the problem with individual ICCs of the first type

$$\begin{array}{ll} \min\limits_{x} & cx \\ \text{s.t.} & \mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] \leq \beta_i, \ i \in I \\ & x \in X. \end{array}$$

We recognize the Lagrangian problem

$$\begin{array}{ll} \min\limits_{x} & cx + \sum\limits_{i \in I} \lambda_i \mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] \\ \text{s.t.} & x \in X, \end{array}$$

defined for $\lambda \geq 0$, as a simple recourse model.

Also for the problem with ICCs of the second type, with $\alpha_i \leq 1/2$, $i \in I$,

$$\begin{array}{ll} \min\limits_{x} & cx \\ \text{s.t.} & \mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] \leq \alpha_i \mathbb{E}_\omega \left[|\eta_i(x,\omega)|\right], \ i \in I \\ & x \in X, \end{array}$$

writing the ICC constraint as in (9), we find that the Lagrangian problem, defined for $\lambda \geq 0$,

$$\begin{array}{ll} \min\limits_{x} & cx + \sum\limits_{i \in I} \left[(1-\alpha_i)\lambda_i \mathbb{E}_\omega \left[\eta_i(x,\omega)^-\right] - \alpha_i \lambda_i \mathbb{E}_\omega \left[\eta_i(x,\omega)^+\right]\right] \\ \text{s.t.} & x \in X, \end{array}$$

is a simple recourse problem.

Remark 4.3 Note that the condition $\alpha_i \leq 1/2$, $i \in I$, which ensures convexity of this ICC feasible set (see Lemma 2.2), corresponds to the well-known condition for sufficiently expensive recourse for SR models, which here reads $(1-\alpha_i)\lambda_i + (-\alpha_i\lambda_i) \geq 0$, $i \in I$.

The Lagrangian problem corresponding to a joint ICC problem is again a recourse problem, see [12], but in this case the recourse structure is complete but not simple.

Given the mathematical relation between individual ICC models and simple recourse models, it is not surprising that the results which motivated Algorithm 3.1 also can be used to develop a special purpose algorithm for simple recourse models. This is the subject of the next section.

5 Simple recourse with random technology matrix

Consider the simple recourse model with random technology matrix (and random right-hand side vector)

$$\begin{array}{ll} \min\limits_{x} & cx + \mathcal{Q}(x) \\ \text{s.t.} & x \in X \end{array}$$

where $\mathcal{Q}(x) := \mathbb{E}_\omega[v(h(\omega) - T(\omega)x)]$, $x \in \mathbb{R}^n$, is the expected value function, and

$$\begin{array}{lll} v(u) := & \min\limits_{y} & q^+y^+ + q^-y^- \\ & \text{s.t.} & y^+ - y^- = u \qquad\qquad u \in \mathbb{R}^m, \\ & & y^+ \in \mathbb{R}^m_+,\ y^- \in \mathbb{R}^m_+ \end{array}$$

is the value function of the second-stage problem. Under the usual assumption that $q_i^+ + q_i^- \geq 0$, $i = 1, \ldots, m$, the function v is finite and convex. The same properties hold for the expected value function $\mathcal{Q}$ under the additional assumption that $\mathbb{E}_\omega[|\omega|]$ is finite.

For the special case with deterministic technology matrix T, i.e., only the right-hand side vector $h(\omega)$ is random, several very efficient algorithms are known (see e.g. [10]). In particular, if ω is a discrete random vector, it is well known that this simple recourse problem is equivalent to a small deterministic LP problem. For example, in the model management system SLP-IOR [9] these algorithms are implemented as the solvers SRAPPROX (Kall & Mayer, 1994) and SHOR1 (Shor & Likhovid, 1997).

However, for the general case with random technology matrix, no special purpose algorithms have been available up to now. Consequently, such models have been solved using general complete recourse algorithms, which by definition do not use the special structure of simple recourse models. Almost all of these algorithms are variants of the L-shaped algorithm [23]; they are based on iterative approximation of the convex function $\mathcal{Q}$ by linear optimality cuts. To compute one such cut, the expected value function $\mathcal{Q}$ is evaluated at the current solution, which amounts to solving the (dual of the) second-stage LP problem for each realization of the (discretized) random parameters.

Below we propose a special purpose algorithm for simple recourse models with discretely distributed technology matrix (and right-hand side vector). The algorithm is based on the results of the previous sections.

5.1 Algorithm for SR with random technology matrix

First we consider separability of the SR expected value function $\mathcal{Q}$. To that end, let ω_i denote the subvector of components of ω on which $T_i(\omega)$ and $h_i(\omega)$ actually depend (note that ω_i and ω_j, $i \neq j$, may overlap). Then $(T_i(\omega), h_i(\omega)) = (T_i(\omega_i), h_i(\omega_i))$, $i = 1, \ldots, m$, so that the function $\mathcal{Q}$ is separable and can be written in closed form:

$$\mathcal{Q}(x) = \sum_{i=1}^{m} \mathcal{Q}_i(x), \qquad x \in \mathbb{R}^n, \tag{15}$$

with, for $i = 1, \ldots, m$,

$$\mathcal{Q}_i(x) := q_i^+ \mathbb{E}_{\omega_i}\left[\left(h_i(\omega_i) - T_i(\omega_i)x\right)^+\right] + q_i^- \mathbb{E}_{\omega_i}\left[\left(h_i(\omega_i) - T_i(\omega_i)x\right)^-\right],$$

where the expectation is taken with respect to the marginal distribution of ω_i. Since $(s)^+ - (s)^- = s$, and using the notation $\eta_i(x, \omega_i) := T_i(\omega_i)x - h_i(\omega_i)$ as before, this is equivalent to

$$\begin{array}{rcl} \mathcal{Q}_i(x) & = & q_i^+ \mathbb{E}_{\omega_i}\left[\eta_i(x, \omega_i)^-\right] + q_i^- \mathbb{E}_{\omega_i}\left[\eta_i(x, \omega_i)^+\right] \\ & = & q_i^-(\tau_i x - \mu_i) + \left(q_i^+ + q_i^-\right)\mathbb{E}_{\omega_i}\left[\eta_i(x, \omega_i)^-\right], \end{array}$$

with $(\tau_i, \mu_i) = \mathbb{E}_{\omega_i}\left[\left(T_i(\omega_i), h_i(\omega_i)\right)\right]$.

Assume that ω, and therefore each ω_i, is discretely distributed, with $\Pr\{\omega_i = \omega_i^s\} = p_i^s$, $s \in S_i$. Then, using (7), we obtain

$$\begin{array}{rcl} \mathcal{Q}_i(x) & = & q_i^-(\tau_i x - \mu_i) + \left(q_i^+ + q_i^-\right) \max\limits_{K_i \subset S_i} \sum\limits_{k \in K_i} -p_i^k \eta_i(x, \omega_i^k) \\ & \geq & q_i^-(\tau_i x - \mu_i) + \left(q_i^+ + q_i^-\right) \sum\limits_{k \in K_i} -p_i^k \eta_i(x, \omega_i^k), \quad K_i \subset S_i. \end{array}$$

In particular, defining $K_i^t := \left\{s \in S_i : \eta_i(x^t, \omega_i^s) < 0\right\}$ for any fixed x^t, we find that

$$\begin{aligned} L_i^t(x) &:= q_i^- (\tau_i x - \mu_i) + \left(q_i^+ + q_i^-\right) \sum_{k \in K_i^t} -p_i^k \eta_i(x, \omega_i^k) \\ &= q_i^- (\tau_i x - \mu_i) + \left(q_i^+ + q_i^-\right) \sum_{k \in K_i^t} p_i^k \left(h_i^k - T_i^k x\right) \end{aligned}$$

is a linear lower bound for Q_i, which is sharp at x^t. Consequently,

$$L^t(x) := \sum_{i=1}^{m} L_i^t(x)$$

is a linear lower bound for the expected value function $\mathcal{Q}$, which is sharp at x^t. That is, for arbitrary fixed x^t, $L^t(x)$ is an optimality cut for $\mathcal{Q}$ at x^t as needed in (variants of) the L-shaped algorithm.

We propose to use the optimality cuts $L^t(x)$ in an L-shaped algorithm for simple recourse problems with discretely distributed technology matrix (and right-hand side vector). There are several advantages in using our approach compared to using a general L-shaped algorithm for complete recourse models.

(i) Using the separability of $\mathcal{Q}$, we only need to consider $\sum_{i=1}^{m} S_i$ second-stage problems instead of $\prod_{i=1}^{m} S_i$ (assuming that rows are independent) to compute $\mathcal{Q}$ at each current solution x^t.

(ii) Instead of finding the optimal value of each second-stage problem by solving an LP problem, we can use the closed form (15) for $\mathcal{Q}(x^t)$. The index sets K_i^t, $i = 1, \ldots, m$, and the (coefficients of) the optimality cut $L^t(x)$ follow trivially from this very fast computation.

5.2 Numerical results

A naive version of the proposed algorithm is implemented in Matlab [8], using the LP solver MOSEK [7] to solve the current problems. In the near future, it will be implemented as a solver in the model management system SLP-IOR [9].

We used the Product Mix problem described in [11] to test our algorithm. This model maximizes expected profit from the production of four goods, given probabilistic information on production technology and the available amount of labor, for two independent parts of the production line. The recourse costs consist of payments for casual labor. The Product Mix problem can be specified as

$$\min_{x \geq 0} \sum_{i=1}^{4} -c_j x_j + \mathcal{Q}(x),$$

where $c = (12, 20, 18, 40)$ and

$$\mathcal{Q}(x) = \sum_{i=1}^{2} q_i \mathbb{E}_{\omega_i} \left[\eta_i(x, \omega_i)^+\right],$$

with $q_1 = 5$, $q_2 = 10$, and $\omega_i = (T_{i1}, \ldots, T_{i4}, h_i)$, $\eta_i(x, \omega_i) = \sum_{j=1}^{4} T_{ij} x_j - h_i$ for $i = 1, 2$. The random parameters T_{ij} and h_i, $i = 1, 2$, $j = 1, \ldots, 4$, are all independently distributed according to Table 4.

To construct instances of this test problem that are suitable for the proposed algorithm, each of the continuous distributions is approximated, independently of the others, by a discrete distribution with k realizations, for $k \in \{2, 3, 4, 5, 10\}$. For each given k, there are

$i = 1$	$i = 2$
$T_{11} \sim \mathcal{U}(3.5, 4.5)$	$T_{21} \sim \mathcal{U}(0.8, 1.2)$
$T_{12} \sim \mathcal{U}(8, 10)$	$T_{22} \sim \mathcal{U}(0.8, 1.2)$
$T_{13} \sim \mathcal{U}(6, 8)$	$T_{23} \sim \mathcal{U}(2.5, 3.5)$
$T_{14} \sim \mathcal{U}(9, 11)$	$T_{24} \sim \mathcal{U}(36, 44)$
$h_1 \sim \mathcal{N}(6000, 100^2)$	$h_2 \sim \mathcal{N}(4000, 50^2)$

Table 4: Distributions of the Product Mix problem ($\mathcal{U}$ and $\mathcal{N}$ denote uniform and normal distributions, respectively).

$S_i = k^5$ realizations for both rows, so that the total number of realizations to consider is $2 \times k^5$ for the proposed algorithm (PA). For general complete recourse algorithms, which do not profit from the separability of $\mathcal{Q}$, the total number of realizations is k^{10}.

In Table 5 the CPU times used by the proposed algorithm are compared to those of the solver DAPPROX v1.0 (Kall & Mayer, 2001), which performed best of all solvers currently available in SLP-IOR. Each instance was solved using a relative accuracy tolerance of 10^{-5} (default for DAPPROX).

k	PA	# cuts	DAPPROX
2	0.79	12	1.94
3	1.07	16	11.07
4	1.24	18	37.49
5	1.48	20	133.12
10	6.72	22	595.90

Table 5: Results in CPU seconds

Remark 5.1 The actual relative accuracy obtained by the proposed algorithm was better than 10^{-8} for all instances with $k \leq 5$. If the relative accuracy tolerance is decreased to 10^{-8} (lower bound for DAPPROX), the cpu time of the proposed algorithm for the instance with $k = 10$ increases to 7.91 seconds (using 26 cuts). In contrast, the cpu time needed by DAPPROX for the larger instances increases substantially: for $k = 5$ it becomes 283.62, and for $k = 10$ even 5559.45 seconds.

We conclude that our special purpose algorithm performs much better than the available complete recourse solvers, at least on this small test problem. In fact, the results of Table 5 have not been corrected for restarting the solver MOSEK (see Section 3.3); these results therefore underestimate the (relative) performance of our algorithm.

6 Summary and conclusions

We presented reduced forms for three types of integrated chance constraints (ICC), for the case that the underlying distribution is discrete. Based on these reduced forms, we proposed an algorithm to solve such ICC problems. Initial numerical results indicate that our algorithm is quite efficient.

An important motivation for this research was the need to find a practical way to include chance constraints (CC) – or, indeed, other variants of one-period risk constraints – in multi-stage recourse models (see [6]), which necessarily make use of discrete distributions (scenarios). We have shown that ICCs are well-suited for this purpose, because they give

rise to convex feasible sets; moreover, they come up in a natural way as the LP relaxation of the mixed-integer LP formulation of traditional chance constraints.

As a by-product, we obtained a special purpose algorithm for simple recourse (SR) models with random technology matrix (and right-hand side vector), again for the case with discrete distributions. Being the first algorithm for this class of problems, it appears to be orders of magnitude faster than existing complete recourse solvers, at least for our small test problem.

We have shown that ICC models are closely related to several other model types. In addition to the CC and SR models mentioned above, we have discussed the relation to conditional surplus-at-risk models. Therefore, one could say that ICC models take a central place in the family of stochastic programming models.

Also for this reason ICC models provide an interesting subject in a stochastic programming course, as we have experienced for several years. In addition, just like SR models, they are relatively easy to understand and specify. The fast algorithms proposed in this paper allow to 'play' with risk parameter values, so that students may develop some feeling for these models (and combined versions). Obviously, this computational efficiency is also very valuable in applications. It not only allows to solve problems of realistic size, but also to obtain additional information from sensitivity analysis with respect to the risk parameters.

Finally, the results reported in this paper have inspired research on convex approximations of traditional CC constraints with discrete distributions. Promising initial results have been obtained by using ICC constraints in combination with special approximations of the underlying discrete distributions, much like the approach that we used for integer recourse models [13, 21]. The results of this line of research will be published in a subsequent paper.

References

[1] F. Andersson, H. Mausser, D. Rosen, and S. Uryasev. Credit risk optimization with conditional value-at-risk criterion. *Math. Program.*, 89(2, Ser. B):273–291, 2001. Mathematical programming and finance.

[2] J.R. Birge and F.V. Louveaux. *Introduction to Stochastic Programming*. Springer Verlag, New York, 1997.

[3] D. Dentcheva, A. Prékopa, and A. Ruszczyński. Concavity and efficient points of discrete distributions in probabilistic programming. *Math. Program.*, 89(1, Ser. A):55–77, 2000.

[4] C.L. Dert. *Asset Liability Management for Pension Funds, A Multistage Chance Constrained Programming Approach*. PhD thesis, Erasmus University, Rotterdam, The Netherlands, 1995.

[5] S.J. Drijver, W.K. Klein Haneveld, and M.H. van der Vlerk. Asset liability management modeling using multistage mixed-integer stochastic programming. Research Report 00E52, SOM, University of Groningen, `http://som.rug.nl`, 2000.

[6] S.J. Drijver, W.K. Klein Haneveld, and M.H. van der Vlerk. ALM model for pension funds: numerical results for a prototype model. Research Report 02A??, SOM, University of Groningen, `http://som.rug.nl`, 2002.

[7] EKA Consulting ApS. The MOSEK optimization toolbox for MATLAB version 2.0 (build 19) User's manual. `www.mosek.com`, 2002.

[8] The MathWorks inc. Matlab: the language of technical computing, version 5.3. `www.mathworks.com`, 1999.

[9] P. Kall and J. Mayer. SLP-IOR: An interactive model management system for stochastic linear programs. *Math. Program.*, 75(2):221–240, 1996.

[10] P. Kall and S.W. Wallace. *Stochastic Programming*. Wiley, Chichester, 1994.

[11] A.J. King. Stochastic programming problems: examples from the literature. In Yu. Ermoliev and R.J-B. Wets, editors, *Numerical techniques for stochastic optimization*, chapter 30, pages 543–567. Springer-Verlag, Berlin, 1988.

[12] W.K. Klein Haneveld. *Duality in stochastic linear and dynamic programming*, volume 274 of *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag, Berlin, 1986.

[13] W.K. Klein Haneveld, L. Stougie, and M.H. van der Vlerk. Convex approximations for simple integer recourse models by perturbing the underlying distribution. Research Report 97A19, SOM, University of Groningen, `http://som.rug.nl`, 1997.

[14] W. Ogryczak and A. Ruszczyński. Dual stochastic dominance and related mean–risk models. Research Report 10, RUTCOR, Rutgers University, Piscataway, NJ, 2001.

[15] G.Ch. Pflug. Some remarks on the value-at-risk and the conditional value-at-risk. In S.P. Uryasev, editor, *Probabilistic constrained optimization*, pages 272–281. Kluwer Acad. Publ., Dordrecht, 2000. Nonconvex Optimization and its Applications, 49.

[16] A. Prékopa. *Stochastic Programming*. Kluwer Academic Publishers, Dordrecht, 1995.

[17] R.T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *The Journal of Risk*, 2(3):21–41, 2000.

[18] S. Sen. Relaxations for probabilistically constrained programs with discrete random variables. *Oper. Res. Lett.*, 11(2):81–86, 1992.

[19] S.P. Uryasev, editor. *Probabilistic constrained optimization*. Kluwer Academic Publishers, Dordrecht, 2000. Methodology and applications.

[20] M.H. van der Vlerk. Stochastic programming bibliography. World Wide Web, `http://mally.eco.rug.nl/biblio/stoprog.html`, 1996-2002.

[21] M.H. van der Vlerk. Convex approximations for complete integer recourse models. Research Report 02A21, SOM, University of Groningen, `http://som.rug.nl`, 2002.

[22] M.H. van der Vlerk. On multiple simple recourse models. Research Report 02A06, SOM, University of Groningen, `http://som.rug.nl`, 2002.

[23] R. Van Slyke and R.J-B. Wets. L-shaped linear programs with applications to control and stochastic programming. *SIAM Journal on Applied Mathematics*, 17:638–663, 1969.